Comparison of Performance Analysis using Different Neural Network and Fuzzy Logic Models for Prediction of Stock Price

Sugandha Saha

ROURKELA

NATIONAL INSTITUTE OF TECHNOLOGY

Department of Computer Science and Engineering

National Institute of Technology Rourkela

Rourkela-769 008, Odisha, India

Comparison of Performance Analysis using Different Neural Network and Fuzzy Logic Models for Prediction of Stock Price

Thesis submitted in partial fulfillment of the requirements for the degree of

Master of Technology

in

Computer Science and Engineering

(Specialization: Software Engineering)

by

Sugandha Saha

(Roll No. 211CS3298)

under the supervision of

Prof. S. K. Rath

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela, Odisha, 769 008, India

June 2013

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Odisha, India.

Certificate

This is to certify that the work in the thesis entitled ***Comparison of Performance Analysis using Different Neural Network and Fuzzy Logic Models for Prediction of Stock Price*** by ***Sugandha Saha*** is a record of an original research work carried out by her under my supervision and guidance in partial fulfillment of the requirements for the award of the degree of Master of Technology with the specialization of Software Engineering in the department of Computer Science and Engineering, National Institute of Technology Rourkela. Neither this thesis nor any part of it has been submitted for any degree or academic award elsewhere.

Place: NIT Rourkela
Date: June 3, 2013

(Prof. S. K. Rath)
Professor, CSE Department
NIT Rourkela, Odisha

Acknowledgment

I am grateful to numerous local and global peers who have contributed towards shaping this thesis. At the outset, I would like to express my sincere thanks to Prof. S. K. Rath for his advice during my thesis work. As my supervisor, he has constantly encouraged me to remain focused on achieving my goal. His observations and comments helped me to establish the overall direction of the research and to move forward with investigation in depth. He has helped me greatly and been a source of knowledge.

I extend my thanks to our HOD, Prof. A. K. Turuk for his valuable advices and encouragement.

I am really thankful to Mrs. Sangeeta Das (Didi), Mr. Manajit Das, (Jiju), Ajay Saha (Brother) and Sanjay Howlader for standing beside me all the time I required any type of support.

I am also thankful to all my friends, especially the PUZZLE group members and hostel wing-mates, seniors and Ph.D scholars for their help and support. My sincere thanks to everyone who has provided me with kind words, a welcome ear, new ideas, useful criticism, or their invaluable time, I am truly indebted.

I must acknowledge the academic resources that I have got from NIT Rourkela. I would like to thank administrative and technical staff members of the Department who have been kind enough to advise and help in their respective roles.

Last, but not the least, I would like to dedicate this thesis to my Ma, Baba and Mr. Aniruddha Saha, for their love, patience, and understanding.

Sugandha Saha
Roll: 211CS3298

Abstract

Analysis and prediction of stock market is very interesting as this helps the financial experts in decision making and in turn profit making. In this thesis simple feed forward neural network (FFNN) model is initially considered for stock market prediction and its result is compared with Radial basis function network (RBFN) model, fuzzy logic model and Elman network model. A FFNN model can fit into any finite input-output mapping problem where the FFNN consists of one hidden layer and enough neurons in the hidden layer. RBFN are the Artificial Neural Networks (ANN) in which Radial Basis Functions (RBF) are used as activation functions. In this thesis, Levenberg-Marquardt Backpropagation algorithm is used to train the data for both FFNN and Elman network. For Fuzzy Logic, Sugeno type Fuzzy Inference System (FIS) is used to model the prediction process. Different Clustering methods are used to find the optimal parameters of RBF. These techniques were tested with published stock market data of National Stock Exchange of India Ltd. for validation.

Keywords: *Clustering; Elman Network; Feed forward neural network; Fuzzy logic; Prediction; Radial basis function; Stock price*

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Chapter 1

Introduction

1.1 Introduction

A stock market is a public market for companies or for people to raise money. Stock market helps companies to buy or sell their shares. The price of shares depends upon the demand and supplies of shares. This process of buying and selling of shares is called trading; only the Listed Companies are allowed to carry out trading.

Stock market prediction is the process of trying to determine the future stock value of a company. The successful prediction of a stock's future price could yield significant profit. Stock price movements are governed by the theories random walk hypothesis and efficient-market hypothesis [1] [2].

The Forecasters of stock market focus on developing approaches which successfully forecast/predict stock prices using well defined trading strategies. A successful prediction model is the one which works with best accuracy having minimum input requirements and least complex model. Investors and government organizations rely on forecasting tools to guard against risks and to monitor market fluctuations. For researchers, these serve as a reference for studies of financial issues like pricing of financial derivatives and portfolio selection.

Generally the stock traders nowadays depend on Intelligent Trading Systems which help them in predicting prices based on various situations and conditions, which in turn help them in making instantaneous investment decisions. Stock market values are considered to be very dynamic and susceptible to quick changes because of the underlying nature of the financial domain and in part because of the mix of known parameters (Previous Day's Closing Price, P/E Ratio etc.) and some other factors (like Election Results, Rumors, climate etc.) [3]. An intelligent trader would predict the stock price and buy a stock before the price of stock rises, or sell it before its value declines. It is hard to replace the expertise that an experienced trader has gained from his experience but an accurate prediction algorithm can directly result into high profits for investment firms, individual professionals, which indicates a direct relationship between the accuracy of the prediction algorithm and the profit made from using the algorithm.

The National Stock Exchange (NSE) of India was set up in 1991 by Government of India on the recommendation of Pherwani Committee. It was incorporated in November 1992 as a tax-paying company. Then it was recognized as a stock exchange in April 1993 under the Securities Contracts (Regulation) Act, 1956.

Many researchers in the past have applied various statistical and soft computing techniques to predict the movements of various stock indices like NASDAQ, Standard & Poor (S&P) 500, Dow Jones Industrial Average (DJIA) index etc. In this thesis prediction techniques are applied to the data of NSE ltd.

1.2 Theories of Stock Market prediction

Stock market has been studied over years to extract useful patterns and predict their movements. Stock market prediction always had an appeal for researchers. In this section, the two most important theories in stock market prediction are explained. The first one is efficient market hypothesis (EMH) introduced in 1964 by Fama and the second one is random walk theory [4].

1. Efficient Market Hypothesis (EMH)

   The EMH states that any form of information cannot be used for generating extraordinary profits from the stock market, as the stock prices always fully reflect all available information. Any type of new information which arises is quickly and efficiently absorbed into the price of the stock. As the way EMH is defined, it is obvious that the result obtained in this work has a direct implication on the validity of the EMH. The efficient market hypothesis (EMH) states that the current stock price reflects the assimilation of all the information available. The EMH hypothesizes that the future stock price is completely unpredictable given the past trading history of the stock. This means no prediction of future change in the price can be made by the historic information. As new information enters the system the unbalanced stock is immediately discovered and quickly eliminated by the correct change in the price.

The EMH exists in three forms, depending on the information which is used for making predictions *weak EMH, semi-Strong EMH, strong EMH.*

In weak EMH, any information acquired from examining the stocks history is immediately reflected in the price of the stock . The "weak" form of EMH states that past stock prices cannot be used to predict future stock prices. Only past price and historical information is embedded in the current price. This EMH rules out any form of prediction based on the price data only, as the prices follow a random walk in which successive change has zero correlation [4].

The semi strong form goes a step further by incorporating all historical and currently public information into the stock price. The strong form includes public, private and historical information. Private information such as company's insider information, in the share price.

The strong form of EMH states that nothing can be used to predict future stock prices as all information is already reflected in the current price of the stock. For the research work in academics and for investment professionals and individuals weak and semi-strong form of EMH has been fairly supported.

2. Random Walk Theory

The random walk hypothesis states that stock price movement does not depend on past stock. With the advent of powerful computing infrastructure (hardware and software), trading companies builds efficient algorithmic trading systems that can exploit the underlying pricing patterns when a huge amount of data-points are available. Clearly with huge datasets available on hand, the machine learning techniques can challenge the EMH. Random walk theory is very much similar to semi-strong EMH where all public information is assumed to be available to everyone. But random walk theory claims that even with such information, future prediction is not so effective.

1.3 Prediction Methods

The prediction of the stock market is an interesting task for researchers. In the literature, number of methods are applied to accomplish this task. The prediction methods use various approaches, from highly informal ways (e.g. the study of a chart with the fluctuation of the market) to more formal ways (e.g. linear or non-linear regressions).

The prediction methods on the basis of type of data and the type of tool that each method is using to predict the market are categorized as:

1. Technical Analysis Methods
2. Fundamental Analysis Methods
3. Traditional Time Series Prediction Methods
4. Machine Learning Methods

The common thing between these techniques is that they are used to predict and thus can have benefit from the market's future behavior.

1.3.1 Technical Analysis

"Technical analysis method - used for predicting the appropriate time to buy or sell the shares of a stock".

Technical analysis is approach to stock investing where the past stock price are studied, using charts as the primary tool. Based on mining rules and patterns from the past prices of stocks which are called mining of financial time series.

The basic principles of technical analysis include concepts such as the trending nature of prices, confirmation and divergence, and the effect of traded volume.

Technical analysis is based on numeric time series data and tries to forecast stock market using indicators of technical analysis. It is based on widely accepted hypothesis, all reactions of the market to all news are contained in real time prices of stocks. Technicians utilize charts and modeling techniques to identify trends in price and volume and Analysts rely on historical data in order to predict future outcomes. This method deals with the determination of the stock price based on the past patterns of the stock using time-series analysis, performed by the technical analysts [4].

Idea of technical analysis indicates that share prices move in trends dictated by the constantly changing attributes of investors in response to different forces. The major point of criticism is that the extraction of trading rules from the study of charts is highly subjective therefore different analysts might extract different trading rules by studying the same charts. It is possible to use this method to predict the market on daily basis but this approach is not followed in this thesis due to its subjective character.

1.3.2 Fundamental Analysis

"Fundamental analysis is the technique of applying the tenets of the firm foundation theory to the selection of individual stocks".

Fundamental analysis investigates the different factors that affect supply and demand. The goal is to gather the information and act before the information is incorporated in the stock price. The lag time between any event and its resulting market response provides a trading opportunity.

Fundamental analysis is based on economic data of companies and tries to forecast markets using economic data that companies have to publish regularly for example annual and quarterly reports, balance sheet, income statements [4]. This is performed by the fundamental analysts. The analysts make their decisions based on the past performance of the company, the earnings forecast etc [4].

This method is concerned more with the company rather than the actual stock.

When applying machine learning and data mining to stock market data, the interest is more in doing technical analysis to see if the algorithm can accurately learn the underlying patterns in the stock time series. Machine learning can also play a major role in evaluating and prediction the performance of the company and other similar parameters helpful in fundamental analysis [5]. In fact, the most successful automated stock prediction and recommendation systems use some sort of a hybrid analysis model involving both Fundamental and Technical Analysis [4].

This type of analysis is not possible to fit in our objectives of study because the data it uses in order to determine the intrinsic value of an asset does not change on daily basis. Hence, fundamental analysis is helpful for predicting the market only in a long-term basis.

1.3.3 Traditional Time Series Prediction

In Traditional Time Series Prediction, the model analyzes historic data and attempts to approximate future values of a time series as a linear combination of these historic data. There are two basic types of time series forecasting: *univariate* (simple regression) and *multivariate* (multivariate regression). These regression models are the most common tools used in econometrics to predict time series. The way they are applied in practice is that firstly a set of factors that influence (or more specific is assumed that influence) the series under prediction is formed.

1.3.4 Machine Learning Methods

The term "Machine learning" includes many inductive learning methods. These methods are used to generate an approximation of the underling function that generated the data using some set of sample data. The main inteerst is to draw conclusions from these samples in such way that when unseen data are presented to a model it is possible to infer the to-be explained variable from these data. In this thesis some machine learning techniques are used for stock market prediction.

1.4 Motivation

The major motivation of our work includes:

- Financial gain is the first interest. For the dynamic market place any system that can consistently pick winners and losers would make the owner of the system very wealthy. Thus, individuals, researchers, investment professionals,and investors are looking for superior system which will yield them high returns.

- Interest is also for the research and financial communities. In the Efficient Market Hypothesis (EMH) it has been proposed that markets are efficient and in that opportunities for profit are discovered so quickly that they cease to be opportunities.

  The EMH states that no system can continually beat the market because if this system becomes public, everyone will use it personally, thus negating potential gain of the system. There has been an ongoing debate about the validity of the EMH, some researchers attempted to use different techniques to validate their claims. There has been no consensus on the validity of EMH, but the market observers tend to believe in the weaker forms of EMH, and thus are often unwilling to share proprietary investment systems.

- Many researchers have used different techniques and generated new technologies and methodologies to predict stock market price, but they never suggest which technology is better. This thesis focuses on comparing different neural network and fuzzy logic models, to determine which technique better suites.

1.5 Organization of thesis

The thesis is organized as follows: Chapter 2 describes the literature review done for this thesis. Chapter 3 discusses about the different techniques used for the prediction of stock price and how the implementation is done. In Chapter 4 all the prediction techniques discussed in chapter 3 are compared. Finally chapter 5 concludes with the summary of work done and discuss the future work.

Chapter 2

Literature Review

Chapter 2

Literature Review

In literature, different sets of input variables are used for stock market prediction. In fact, different input variables are used to predict the same set of stock market data. Some researchers used inputs from a single time series and some considered the inclusion of heterogeneous market information and macroeconomic variables.

A lot of research has been done and models based on a range of intelligent soft computing techniques are developed over the last two decades. This section describes briefly some of the work that has already been done in the field of stock price prediction.

In 1990, technology major Fujitsu and investment company, Nikko Securities joined hands to develop a stock market prediction system for TOPIX, Tokyo based stock index, using modular neural network architecture [6]. Various economic and technical parameters were taken as input to the modular neural network consisting of multiple MLP used in parallel.

In 1993 research was done on the effect of change of network parameters of the model using artificial neural network (ANN) with Backpropagation on the stock price prediction problem [7]. The paper gives information about the role of the learning rate, momentum, activation function and the number of hidden neurons for prediction of stock market.

There are fuzzy logic based application models for the stock market prediction as well. A fuzzy logic forecast support system was used to predict the stock prices using parameters such as inflation, GNP growth, interest rate trends and market valuations [8]. The paper specifies that due to the model-based approach with knowledge management and knowledge accumulation the potential benefits of a fuzzy logic forecast support are better decision making.

The Probabilistic Neural Network (PNN) has been employed to stock prediction [9] in addition to ANN. This model is used to draw up a conservative thirty day stock price prediction of a specific stock: Apple Computers Inc. The PNN are not popular among forecasters due to their bulky nature owing to the large training data.

A hybrid model that integrates GA based fuzzy logic and ANN [10] has been proposed. The model involves quantitative factors (technical parameters) and qualitative factors such as political, environmental and psychological factors. The results shows that the neural network that considers the quantitative and qualitative factors both, excels to the neural network considering only the quantitative factors, both in the clarity of buying-selling points and buying selling performance.

Researchers also used Hidden Markov Models (HMM) approach for forecasting stock price for interrelated markets [11]. HMM, because of its proven suitability for modeling dynamic system was used for pattern recognition and classification problems. It is able to handle new data robustly and also is computationally efficient to develop and evaluate similar patterns. To improve the accuracy and efficiency of forecast the stock market, the author decides to develop hybrid system using AI paradigms with HMM.

More efforts has been made towards the development of fuzzy models for stock market prediction made using Takagi-Sugeno (TS) fuzzy models in 2006 [12]. In this paper the models described are used for effort estimation and stock market prediction using TS fuzzy models. The two steps of the process are 1) the determination of the membership functions in the rule antecedents using the model input data; 2) the estimation of the consequence parameters. Parameters are estimated using least square estimation.

Another hybrid model utilizes the strengths of Hidden Markov Models (HMM), ANN and GA to forecast financial market behavior [13]. The job of the GA is to optimize the initial parameters of HMM. Using ANN, the daily stock prices are transformed to independent sets of values that become input to HMM. The trained HMM is then used to identify and locate similar patterns in the historical data.

The moving average autoregressive exogenous (ARX) prediction model is combined with grey system theory and rough set theory to create an automatic stock market forecasting and portfolio selection mechanism [14].Input to an ARX prediction model for forecast the future trends are the financial data collected automatically every quarter. Using a K means clustering algorithm the data are clustered and then supplied to a RS classification module which selects appropriate investment stocks by a decision-making rules. The advantage of combining different forecasting techniques is to improve the efficiency and accuracy of automatic prediction. The hybrid model provides a highly accurate forecasting performance.

A hybrid forecasting model based on rough sets theory [15] using multi-technical indicators was proposed to predict stock price trends. Four procedures are described such as select the essential technical indicators required, find the popular indicators based on a correlation matrix and use cumulative probability distribution approach to minimize the entropy principle approach. Then finally use RST algorithm to extract linguistic rules and refine the extracted rules to get better forecasting accuracy and stock return using genetic algorithm. The advantage discovered was that this model produces more reliable and understandable forecasting rules based on objective stock data rather than subjective human judgments.

Markov chain concepts were used in fuzzy stochastic prediction of stock indexes [16] to achieve better accuracy and confidence. This paper examined the comparison of ANN and Markov model, has major advantages. It generates high accurate result and with only one input of data. Input used was the first hour's stock index and it lead the prediction of the probable index at any given hour. This approach provides not only improved profit performance but is also used to determine stop-losses with greater confidence.

ARIMA model and vector ARMA model was used with fuzzy time series method for forecasting [17]. Fuzzy time series method performs better forecasting, especially heuristic model in short-term period prediction. The ARIMA model creates small forecasting errors in longer experiment time period. In this paper it is investigated whether the length of the interval will influence the forecasting ability of the models or not.

Adaptive Neuro Fuzzy system controller used to control the stock market process model [18] and also evaluate a variety of stocks. The Efficient Market Hypothesis was used to improve the prediction in short-term stock market trends. The result demonstrates clearly to use the proposed Rate of Return (ROR). The better returns was obtained the investor allocated assets to the risk-free government bonds once the predicted stock return turned negative. This is known as asymmetric outcomes of the stock markets. The second time the investor allocated assets to risk-free government bonds have some positive returns. This means gains from correct prediction and losses from incorrect prediction. The neuro-fuzzy system clearly demonstrates the potential for financial market prediction.

Chapter 3

Proposed Work

Chapter 3

Proposed Work

3.1 Data Set

The real stock market data from NSE [19] of India Ltd has been used for the prediction models. The proposed techniques have been experimented with four data sets i.e., Stock data of State Bank of India (SBI), HDFC Bank, Steel Authority of India Ltd. (SAIL) and Jindal Steel Power Ltd. (JSP Ltd.). Each data set is divided into two parts, one is used for training and other is used for testing. Table-3.1 shows the division of the data set into training data and test data. Input used are the ten days opening price and the output is the next day's (eleventh day's) opening price.

Table 3.1: Information of training and test data

Stock Name	**Training Data**		**Test Data**	
	From	**To**	**From**	**To**
SBI	3/9/2007	2/9/2011	3/9/2011	31/12/2012
HDFC	3/9/2007	2/9/2011	3/9/2011	31/12/2012
JSP Ltd.	3/9/2008	2/9/2011	3/9/2011	31/12/2012
SAIL	3/9/2008	2/9/2011	3/9/2011	31/12/2012

3.1.1 Data Preprocessing

As the range of data values is very large, The data has been normalized. Normalization transforms measures of magnitude (counts or weights) into measures of intensity. It is the process of creating the shifted and scaled versions of statistics; this is done because the normalized values eliminate the effects of certain gross influences of the data. To normalize a set of data, the original data range is mapped into another scale. Normalization is needed to pre-process data so as to ease the algorithm's job i.e. it helps to bring the data closer to the requirements of the algorithms. Variables can be normalized (to unit zero mean and unit variable, or to the interval [0, 1]), data elements can be normalized (when all their attributes have the same 'units') and the target variable can be normalized too (using a logarithmic transform). Following are the steps of normalizing data sets:

- *Find out the Minimum ($MinO$) and Maximum ($MaxO$) of original data sets.*
- *Decide the Minimum ($MinN$) and Maximum ($MaxN$) for normalized scale.*
- *Consider a number (A) from the data set.*
- *The Normalized value for the number(A) is given by the formula:*

$$MinN + \frac{(A - MinO)(MaxN - MinN)}{MaxO - MinO} \tag{3.1}$$

3.2 Proposed Prediction Techniques

3.2.1 Feed Forward Neural Network

Introduction

Feed forward neural networks [20,21], trained with a back-propagation learning algorithm, are the most popular neural networks. They are applied to a wide variety of problems. A FFNN consists of neurons, that are ordered into layers. The first layer is called the input layer, the last layer is called the output layer, and the layers between are hidden layers. A single hidden layer FFNN is shown in Figure-3.1. Each neuron in a particular layer is connected with all neurons in

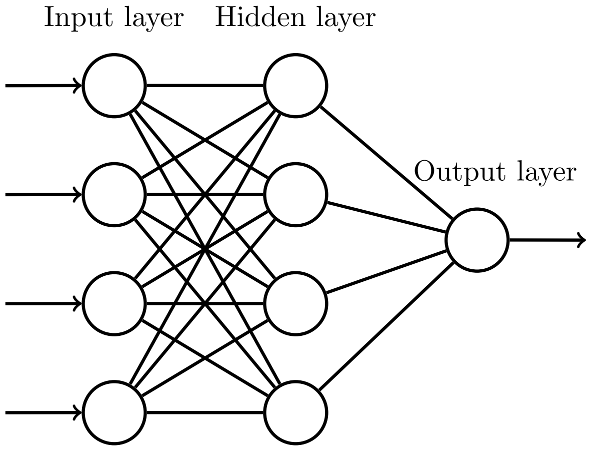

Figure 3.1: A Typical Single Layer FFNN

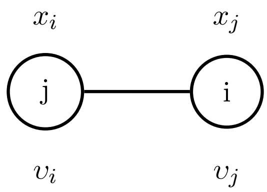

Figure 3.2: Connection between two neurons

the next layer. The connection between the i^{th} and j^{th} neuron is characterized by the weight coefficient w_{ij}. The weight coefficient reflects the degree of importance of the given connection in the neural network. The output of a neuron can be determined by equations 3.2 and 3.3

$$x_i = f(\xi_i) \tag{3.2}$$

$$\xi_i = \upsilon_i + \sum_{j \epsilon \Gamma_i^{-1}} w_{ij} x_j \tag{3.3}$$

where ξ_i is the potential of the ith neuron, $f(\xi_i)$ is the transfer function, Γ is the mapping function of neurons, Γ_i^{-1} consists of all the predecessors of the given neuron i.

Levenberg-Marquardt backpropagation algorithm [22–24] The objective of a training algorithm is to reduce the the global error E given by:

$$E = \frac{1}{P} \sum_{p=1}^{p} E_p \tag{3.4}$$

where P is the number of training patterns, E_p is the error of training pattern p calculated as:

$$E_p = \frac{1}{2}\sum_{i=1}^{N}(O_i - t_i) \tag{3.5}$$

where N total number of output data, o_i is the network output of i^{th} data and t_i is the target output of i^{th} data.

In the training algorithm attempts are made to reduce the global error by adjusting weights and biases.

This algorithm [25] introduces approximation to Hessian matrix (H):

$$H \approx J^T J + \mu I \tag{3.6}$$

where J is the Jacobian matrix, μ is always positive, called combination coefficient and I is the identity matrix. Hence,the update rule of the Levenberg-Marquardt algorithm is given by:

$$w_{k+1} = w_k - (J_k^T J_k + \mu I)^{-1} J_k e_k \tag{3.7}$$

where k is the index of iterations, w is the weight vector, e is the training error. As the combination [25] of the steepest descent algorithm and the GaussNewton algorithm, the Levenberg-Marquardt algorithm switches between the two algorithms during the training process.

Overview of the prediction process

The flow chart of prediction process using FFNN is shown in Figure- 3.3. The steps of prediction using FFNN are:

1. **Data:** Select the required stock market data. Refer Section- 3.1.

2. **Data Preprocessing:** Normalize the required data in the range of 0-1.

3. **Creating a FFNN**: Set the initial parameters and create the required FFNN model.

4. **Training:** Train the model using backpropagation algorithm and set the final parameter values.

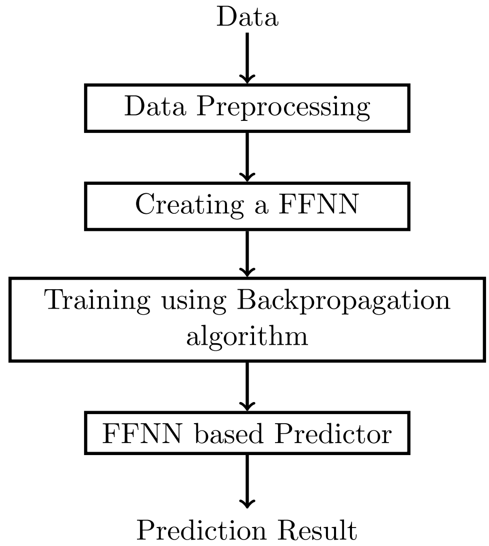

Figure 3.3: Overview of Prediction Process using FFNN

5. **FFNN Predictor:** Now using the final parameters that are set in the previous step the output is found.

6. **Prediction Result:** This is the required output.

Steps Followed for the Prediction of Stock Price using FFNN

- **Step 1:** The training and testing, input and output data are loaded.

- **Step 2:** The data sets are normalized.

- **Step 3:** The FFNN model with 10 neurons in input layer, 10 neurons in the hidden layer, 1 neuron in the output layer is created.

- **Step 4:** The initial parameters are taken as: learning rate=0.01, training ratio = 0.1, validation ratio = 0.1, testing ratio = 0.1, performance goal = 0, epochs = 100.

- **Step 5:** The created FFNN model is trained with the training data.

- **Step 6:** Use the trained FFNN model of previous step with the test input data to get the predicted output.

- **Step 7:** Plot the actual output values and the predicted output values.

- **Step 8:** Root Mean Square Error is computed
- **Step 9:** Magnitude of Relative Error is calculated.
- **Step 10:** Mean of MRE is computed.
- **Step 11:** Accuracy is computed.

3.2.2 Radial Basis Function Network

Introduction

Radial basis function neural network, a variant of ANN emerged in late 80's is a three-layered feed forward network. It consists of input layer, hidden layer and output layer. The input layer contains units of signal source, and the second layer is hidden layer. The number of units on the hidden layer is determined by necessity. The third layer is an output layer which reacts to input model. Movement from input layer to hidden layer is nonlinear and that from hidden layer to output layer is linear. Activation function of the units in hidden layer is RBF, which can be demonstrated by the Figure-3.4.

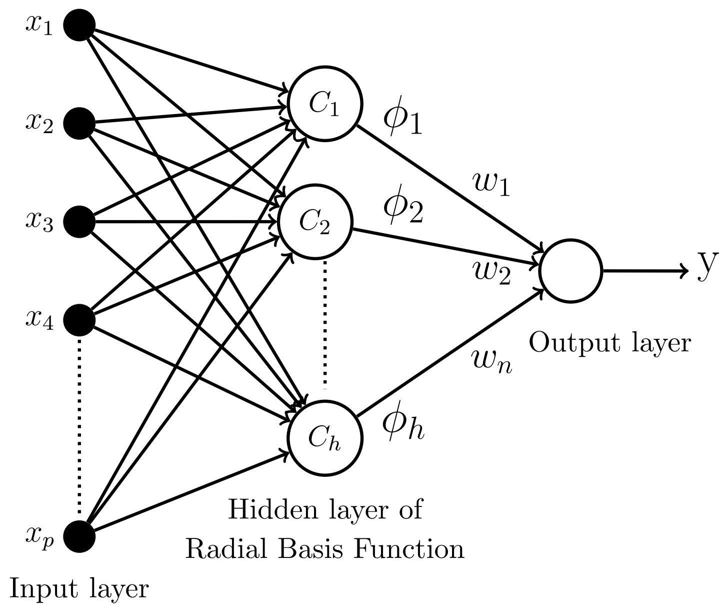

Figure 3.4: RBFN Network

In Figure-3.4, X = $(x_1, x_2, ..., x_p)$ is an p-dimensional vector; and W= $(w_1, w_2, ..., w_n)$ is the weight of output layer. The output of the i^{th} neuron in hidden layer of RBF can be calculated as:

$$q_i = \phi_i(\|X - C_i\|) = exp(-\frac{(\|X - C_i\|)^2}{2\sigma_i^2}) \tag{3.8}$$

where,

σ_i: Width if the receptive field,

ϕ_i: Guassian Activation function, i=1, 2, ..., h,

h: Number of neurons in hidden layer,

C_i: Center of i^{th} activation function and

$\| * \|$: Euclid norm.

The activation of the output layer is linear combination of units.

Clustering Methods To determine the parameters of a RBF network clustering methods can be used. We have used following clustering methods for a different RBF network.

1. **K-means Clustering**:

   K-means [26] developed in 1967 by MacQueen is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result. So, it is better to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is left, the first step is completed and an early groupage is done. At this point k new centroids are recalculated as barycenters of the clusters resulting from the previous step. Using the k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop it can be noticed that that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more. This algorithm aims at minimizing an objective function, a squared error function i.e.

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \| x_i^{(j)} - c_j \|^2 \tag{3.9}$$

where $\| x_i^{(j)} - c_j \|^2$ is a chosen distance measure between a data point $x_i^{(j)}$ and the cluster center c_j, is an indicator of the distance of the n data points from their respective cluster centers.

The algorithm consists of the following steps:

(a) Place K points into the space represented by the objects that are being clustered. These points represent initial group centroids.

(b) Assign each object to the group that has the closest centroid.

(c) When all objects have been assigned, recalculate the positions of the K centroids.

(d) Repeat Steps 2 and 3 until the centroids no longer move. This produces a separation of the objects into groups from which the metric to be minimized can be calculated.

Although it can be proved that the procedure will always terminate, the k-means algorithm does not necessarily find the most optimal configuration, corresponding to the global objective function minimum. The algorithm is also significantly sensitive to the initial randomly selected cluster centers. The k-means algorithm can be run multiple times to reduce this effect.

2. **Fuzzy C-mean Clustering (FCM):**

In traditional clustering approaches partitions are generated; in a partition, each instance belongs to one and only one cluster. The clusters in a hard clustering are therefore disjointed. Fuzzy clustering suggests a soft clustering schema. In this clustering, each pattern is associated with every cluster using some sort of membership function, namely, each cluster is a fuzzy set of all the patterns. More is the membership value; indicate higher confidence in the assignment of the pattern to the cluster. By using a threshold of the membership value for fuzzy partition, hard clustering can be obtained. FCM is the most popular fuzzy clustering method. It is better than the hard K-means algorithm at avoiding local minima, but FCM can also converge to local minima of the squared error criterion.

Fuzzy c-means (FCM) clustering method was developed by Dunn in 1973 and improved by Bezdek in 1981. FCM [27] clustering is based on the minimization of the objective function Q:

$$Q_m = \sum_{i=1}^{N} \sum_{j=1}^{C} q_{ij}^{m} \|x_i - c_j\|^2, 1 \leq m < \infty \tag{3.10}$$

where,

m: is a real number greater than or equal to 1,

q_{ij}: is the degree of membership of x_i in the cluster j,

x_i: is the ith data and

c_j: is the center of the cluster.

Clustering is carried out through an iterative optimization of objective function Q, updating degree of membership q_{ij} and cluster centers c_j by:

$$q_{ij} = \frac{1}{\sum_{k=1}^{C} \left(\frac{\|x_i - c_j\|}{\|x_i - c_k\|} \right)^{\frac{2}{m-1}}} \tag{3.11}$$

$$c_j = \frac{\sum_{i=1}^{N} q_{ij}^{m} . x_i}{\sum_{i=1}^{N} q_{ij}^{m}} \tag{3.12}$$

The steps of the algorithm are:

(a) Initialize $U = [u_{ij}]$ matrix, $U^{(0)}$.

(b) At k-step: calculate the centers vectors $C^{(k)} = [c_j]$ with $U^{(k)}$ from Equation 3.12.

(c) Update $U^{(k)}$, $U^{(k+1)}$ from Equation 3.11

(d) If $\| U^{(k+1)} - U^{(k)} \| < \varepsilon$ then STOP; where ε is a termination criterion between 0 and 1; otherwise return to step (b).

3. **Subtractive Clustering:**

The subtractive clustering technique is proposed by Stephen L. Chiu in 1994. Clustering has been often exercised as a preprocessing input phase used in the design of the RBF neural networks. Subtractive clustering operates by finding the optimal data point to define a cluster center based on the density of the surrounding data points. To use subtractive clustering, four parameters should be preinitialized [28] . These parameters are *Hypersphere cluster radius in data space, Squash Factor, Reject Ratio, Accept Ratio.* Hypersphere cluster radius in data space defines a neighborhood data points outside this radius has little influence on the potential. Squash Factor defines the neighborhood which will have the measurable reductions in potential and it can be calculated as:

$$SquashFactor(\eta) = \frac{r_b}{r_a} \tag{3.13}$$

Reject Ratio specifies a threshold for the potential above which the data point is definitely accepted as a cluster centre. *Accept Ratio* specifies a threshold below which the data point is definitely rejected. Consider a collection of q data points $x_1, x_2, ..., x_q$ where x_i is a vector in the feature space. Without the loss of generality, it is assumed that the feature space is normalized so that all data are bound by unit hypercube. The potential of each data point defines a measure of the data point to serve as a cluster center. The potential for each data point can be calculated by using the following equation.

$$P_i = \sum_{q}^{j=1} exp(-\frac{\|x_i - x_j\|}{(\frac{r_a}{2})^2}) \tag{3.14}$$

where $\|.\|$ denotes the Euclidean distance, and r_a is a positive constant called cluster radius. After the potential of each data point has been calculated, the data point with the highest potential is selected as the first cluster center.

Let x_1^* be the center of the first cluster and p_1^* its potential value. The potential of each data point revised is given by:

$$p_i = p_i - p_i^* exp(-\frac{\|x_i - x_i^*\|}{(\frac{r_b}{2})^2}) \tag{3.15}$$

where η is a positive constant greater than 1 and is called the squash factor. When the potentials of all data points have been revised by Equation 3.16, the data point with the highest remaining potential is selected as the second cluster center. In general, after the L_{th} cluster center has been obtained, the potential of each data point is revised as follows:

$$p_i = p_i - p_L^* exp(-\frac{\|x_i - x_L^*\|}{(\frac{r_b}{2})^2}) \tag{3.16}$$

where x_L^* is the centre of the L_{th} cluster and p_L^* is its potential value.

Overview of the prediction process

The flow chart of prediction technique using RBFN is shown in Figure-3.5 The

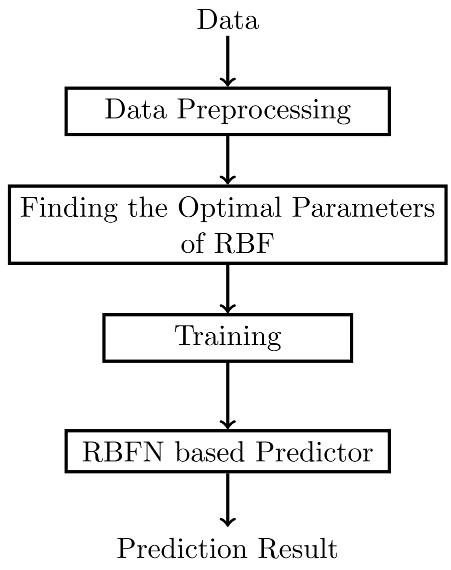

Figure 3.5: Overview of Prediction Process using RBFN

steps of prediction using RBFN are:

1. **Data:** Select the required stock market data. Refer Section- 3.1.

2. **Data Preprocessing:** Normalize the required data in the range of 0-1.

3. **Finding the optimal parameters of RBF**: Use one of the clustering method described in section- 3.2.2 to find the parameters of RBF.

4. **Training:** Train the parameters and set the final parameter values.

5. **RBFN Predictor:** Now using the final parameters that are set in the previous step to find the required output.

6. **Prediction Result:** This is the required output.

Steps Followed for the Prediction of Stock Price using RBFN

- **Step 1:** The training and testing, input and output data are loaded.

- **Step 2:** The data sets are Normalized.

- **Step 3:** The training input data is clustered using each of Kmeans, Fuzzy C-mean and Subtractive clustering methods.

- **Step 4:**'σ' is calculated for gaussian function.

$$\sigma = \frac{max.DistanceBetweenAnyTwoCenters}{\sqrt{NumberOfCenters}} \tag{3.17}$$

  The activation function of the hidden neuron is:

$$\phi_i(\|X - C_i\|) = exp(-\frac{(\|X - C_i\|)^2}{2\sigma_i^2}) \tag{3.18}$$

- **Step 5:** The weights are calculated using PseudoInverse function

  $w = (\Phi^T \Phi)^{-1} \Phi^T Y'$, $\Phi = (\phi_1, \phi_2, ..., \phi_h)$, Set of radial basis functions and Y', actual output of training dataset.

- **Step 5:** The weights calculated in previous step and the testing data are used to get the final output.

- **Step 6:** Root Mean Square Error is computed.

- **Step 7:** Magnitude of Relative Error is computed.

- **Step 8:** Mean of MRE is calculated.
- **Step 9:** Accuracy is calculated.

3.2.3 Fuzzy Logic

Introduction

Fuzzy sets were introduced by L. A. Zadeh (1965) as a means of representing and manipulating data that was not precise, but rather fuzzy. Fuzzy logic provides an inference morphology that enables approximate human reasoning capabilities to be applied to knowledge-based systems.

Fuzzy system consists of three main components: fuzzication process, inference from fuzzy rules and defuzzication process. Among various fuzzy models, the model introduced by Takagi, Sugeno and Kang (TSK fuzzy system) [29] [30] is more suitable for sample-data based fuzzy modeling, because it needs less rules. Each rule's consequence with linear function can describe the input-output mapping in a large range, and the fuzzy implication used in the model is also simple.

Overview of the prediction process

The flow diagram of the prediction technique using fuzzy logic is shown in Figure-3.6

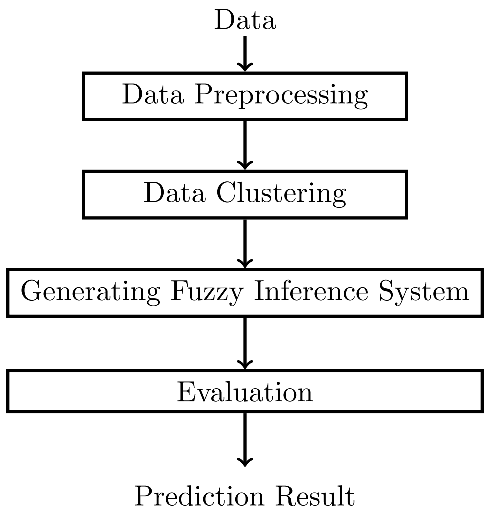

Figure 3.6: Overview of Prediction Process using Fuzzy Logic

The steps of prediction using Fuzzy Logic are:

1. **Data:** Select the required stock market data. Refer Section- 3.1.
2. **Data Preprocessing:** Normalize the required data in the range of 0-1.
3. **Data Clustering:** Use the subtractive clustering method described in section- 3.2.2 to find clusters and cluster centers.
4. **Generating Fuzzy Inference System:** Using the centers draw some rules and generate a fuzzy inference system.
5. **Evaluation:** Use the fuzzy inference system to find the required output.
6. **Prediction Result:** This is the required output.

Pseudocode

- **Step 1:** Load the training and testing, input and output data.
- **Step 2:** Generate an initial fuzzy inference system with the input data, output data, radius and the normalization values as inputs.
- **Step 3:** Use the generated fuzzy inference system and testing input data to evaluate the required output.
- **Step 4:** Plot the actual output values and the predicted output values.
- **Step 5:** Calculate root mean square error
- **Step 6:** Calculate the Magnitude of Relative Error.
- **Step 7:** Calculate the Mean of MRE.
- **Step 8:** Calculate the accuracy.

3.2.4 Elman Network

Introduction

Basic Elman network [31] shown in Figure-3.7 consists of input layer, hidden layer, context layer and output layer. Based on the structure of BP network, Elman network adds a feedback layer which is used to maintain former-moment output state of interlayer unit so as to represent internal characteristics. This feedback layer can be thought to be a one step lag operator. Moreover, this feedback connector consists of a group of "connect" units which are used to memorize past state of implicit layer and to be input of implicit units combing with network input in the next moment. Thus it equals to state feedback and this feature enable partial recursive network to own dynamic-memory function which is appropriate for establishing prediction model of temporal series.

The inputs of every layer are all weight sum; the transfer function of implicit layer is still a certain kind of non-linear function which usually is Sigmoid function. In this function, Both the input layer and connection layer are linear functions. These internal state and external input signal are used as the input of interlayer nodes in present moment. The features of Elman recurrent nerve unit network is that via the lag, memory of structure units, the output of interlayer self-connect to the input of interlayer. This kind of self-connection is sensitive to its data of history state. Moreover, this network's internal feedback network, which is advantageous in dynamic process modeling, enhances the ability of its own dynamic information process. Because Elman network can memorize information for future usage, it can not only study virtual domain model, but can also study real domain one.

In addition, because the dynamic characteristics of Elman network is provided by internal connection, it does not need to use states as input or training signal which is the advantage of Elman network comparing to static feed forward network

Overview of the prediction process

1. **Data:** Select the required stock market data. Refer Section- 3.1.

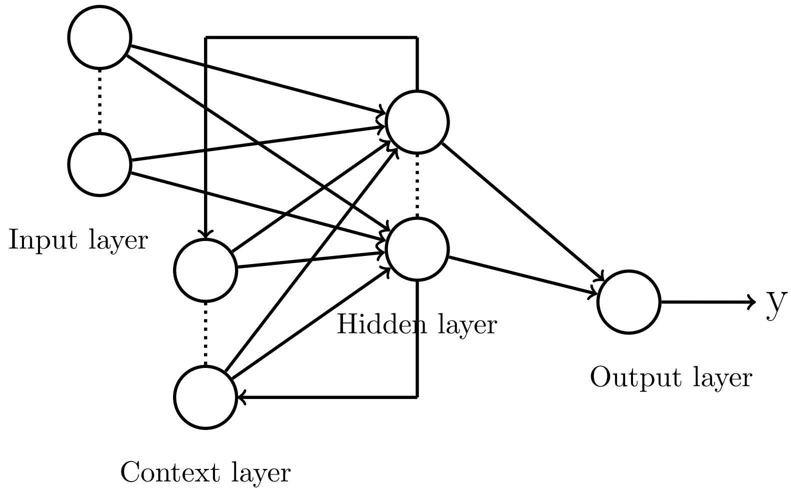

Figure 3.7: Elman Network

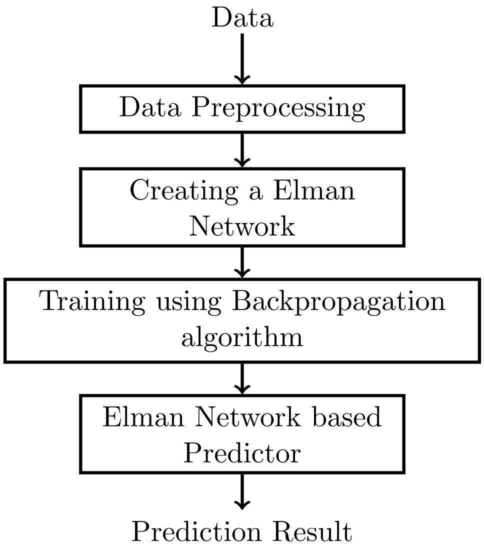

Figure 3.8: Overview of Prediction Process using Elman network

2. **Data Preprocessing:** Normalize the required data in the range of 0-1.

3. **Creating Elman Network**: Set the initial parameters and create the required Elman Network.

4. **Training:** Train the model using backpropagation algorithm and set the final parameter values.

5. **Elman Network Based Predictor:** Now using the final parameters that are set in the previous step the output is found.

6. **Prediction Result:** This is the required output.

Pseudocode

- **Step 1:** Load the training and testing, input and output data.
- **Step 2:** Normalize the data sets.
- **Step 3:** Convert training data of concurrent vectors with a matrix to sequential vectors with a cell array.
- **Step 4:** Create the Elman network model.
- **Step 5:** Set the initial parameters i.e. learning rate, training ratio, validation ratio, testing ratio, performance goal, epochs.
- **Step 6:** Use the created model and the sequential vectors of training data and determine the Shifted inputs (Xs), Initial input delay states (Xi), Initial layer delay states (Ai) and Shifted targets (Ts).
- **Step 7:** Train the network model using Xs, Xi, Ai, Ts and function.
- **Step 8:** Use the trained network model of previous step and *sim* function in matlab with the the testing input data that is converted to sequential vectors to get the output.
- **Step 9:** Convert the sequential vector output to concurrent output, the required output.
- **Step 10:** Plot the actual output values and the predicted output values.
- **Step 11:** Calculate the Root Mean Square Error
- **Step 12:** Calculate the Magnitude of Relative Error.
- **Step 13:** Calculate the Mean of MRE.
- **Step 14:** Calculate the Accuracy.

Chapter 4

Implementation and Result

Chapter 4

Implementation and Results

4.1 Implementation

Data preprocessing, training and testing of the proposed techniques are implemented in Matlab R2012a.

4.1.1 Feed Forward Neural Network

A 10-2-1 architecture i.e., 10 neurons in input layer, 10 neurons in hidden layer and one neuron in output layer is used to create the required FFNN model for prediction [32]. For training Levenberg-Marquardt backpropagation algorithm has been used. Levenberg-Marquardt is the mostly used training algorithm for time series prediction. The parameters considered for training the FFNN are:

- Performance Goal = 0
- Learning Rate = 0.01
- Maximum number of epochs = 100
- Training ratio = 0.8
- Validation ratio = 0.1
- Testing ratio = 0.1

After training a trained network is obtained, network structure is shown in Figure 4.1. Hyperbolic tangent sigmoid transfer function and the linear transfer function

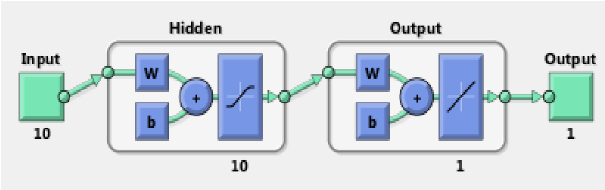

Figure 4.1: FFNN Structure.

are used in hidden layer and output layer respectively of this FFNN. Using the training parameters final weights and bias were found which were used in the testing process.

4.1.2 Radial Basis Function Network

In the training process of RBFN network following parameters are determined:

- The number of hidden layer neurons.
- For each hidden-layer RBF function find the coordinates of the centers.
- The radius (spread) of each RBF function in each dimension.
- The weights that are applied to the RBF function outputs are calculated as:

$$w = (\Phi^T \Phi)^{-1} \Phi^T Y' \tag{4.1}$$

where,

$\Phi = (\phi_1, \phi_2, ..., \phi_h)$, Set of radial basis functions.

Y': Actual output of Training dataset.

For determining the above mentioned training parameters of RBFN network following clustering methods are used:

1. K-means clustering- No. of Clusters (k)= 6.
2. Subtractive clustering- The subtractive clustering parameters were tested for different values and were finally taken as:
   - Radii = 0.3

- Squash factor = 1.25
- Accept ratio = 0.4
- Reject ratio = 0.1

After performing clustering the final weights found were used for the testing process.

3. Fuzzy c-means clustering- No. of Clusters= 6.

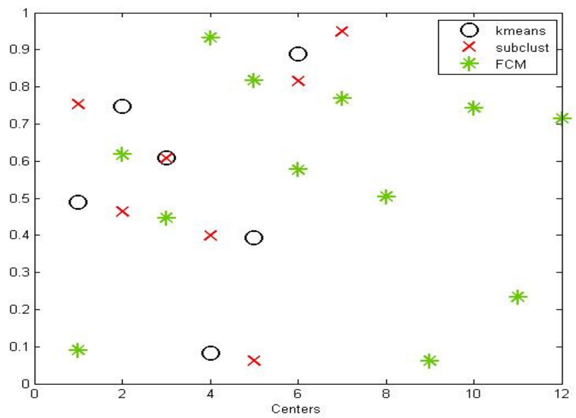

Figure 4.2: Centers found for different Clustering methods for a dataset

The centers found for the dataset of a case i.e., Jindal Steel Plant for all the three clustering methods is shown in Figure-4.2. Similarly for all the data sets, the centers were calculated using the three mentioned clustering methods.

4.1.3 Fuzzy Logic

Consider a system with $x_1(k)$, $x_2(k)$, $x_3(k)$,...$x_n(k)$ are the input variables and y(k) as the output variable, respectively. k is the time sample. The output variable is represented as:

$$y(k) = f(x_1(k), x_2(k), ..., x_n(k)) \tag{4.2}$$

The objective is to find the values of the model output y(k) as a function of past outputs. Fuzzy models of different types can be used to approximate this relationship function f, here Gaussian functions are used as the membership functions. A fuzzy model of a dynamic system consists of a set of rules of the following form:

$$R_i : \textbf{If} x_1(k) is A_1 \textbf{and} ... \textbf{and} x_n(k) is A_n \textbf{then} y(k) is c_i \tag{4.3}$$

Here, 10 inputs are given to the model for a single output and thus there are 10 membership functions. The graph for membership functions are shown in Figure-4.3.

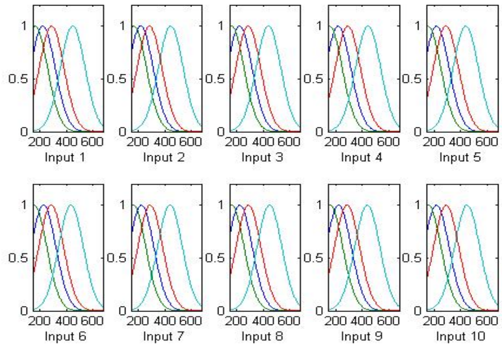

Figure 4.3: Gaussian membership functions

4.1.4 Elman Network

A 10-2-1 architecture i.e., 10 neurons in input layer, 1 neuron in hidden layer and one neuron in output layer with tap delay of 2units is used for the Elman network model. For training again Levenberg-Marquardt backpropagation algorithm is used as in FFNN. The network structure of elman network is shown in Figure-4.4. Here also Hyperbolic Tangent Sigmoid transfer function and the Linear Tranfer function are used as the activation functions in hidden layer and output layer respectively. The parameters considered for training the FFNN are:

- Performance Goal = 0
- Learning Rate = 0.01
- Maximum number of epochs = 100
- Training ratio = 0.8
- Validation ratio = 0.1

- Testing ratio = 0.1

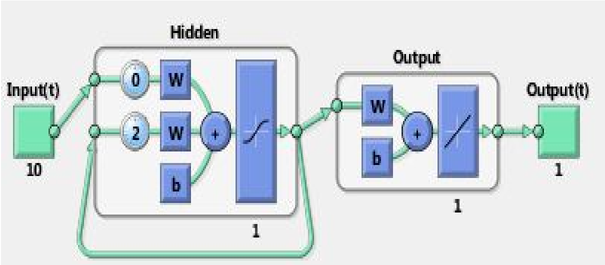

Figure 4.4: elman

4.2 Results

4.2.1 Performance Criteria

1. Root mean square error (RMSE): RMSE is frequently used performance criteria which measures the difference between values predicted by a model or forecaster and the values actually observed. It is the square root of the mean square error, as shown in equation 4.4 given below:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \tag{4.4}$$

where,

N: Number of Data points,

i: i^{th} Data point,

y_i: Actual value,

$\hat{y}_i$: Predicted value.

2. Mean magnitude relative error (MMRE): MMRE is also used to evaluate the performance of any forecasting technique. It measures the difference between forecast and actual value relative to the actual value, as shown in equation 4.5 given below:

$$MMRE = \frac{1}{N}\sum_{i=1}^{N}\frac{|y_i - \hat{y}_i|}{y_i} \tag{4.5}$$

3. Accuracy: Accuracy is the degree of matching between the predictions and the actual data. It is calculated as:

$$Accuracy = 100 - 100 * \frac{\Sigma|X - Y|}{N} \tag{4.6}$$

where,

X: Actual data values,

Y: Predicted Data values,

$|X - Y|$: Absolute value of X-Y and

N: Number of data points in X

4.2.2 Performance Comparison of the Proposed Techniques

The graph for actual vs. predicted values of all the four data sets of each organisation are shown in Figure-4.5, Figure-4.6, Figure-4.7 and Figure-4.8. The performance comparisons of the techniques are shown in Table-4.1, Table-4.2, Table-4.3 and Table-4.4.

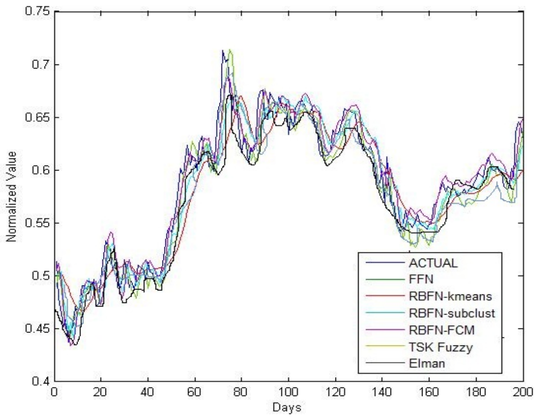

Figure 4.5: Actual Vs. Predicted Values for SBI

Table 4.1: Comparison of prediction techniques for data of SBI

Prediction technique	RMSE	MMRE	Accuracy
FFNN	0.0171	0.0228	98.70
RBFN with Kmeans Clustering	0.0256	0.0343	98.02
RBFN with Fuzzy C-Means Clustering	0.0266	0.0360	98.49
RBFN with Subtractive Clustering	0.0198	0.0258	98.75
TSK Fuzzy	0.0144	0.0178	98.91
Elman Network	**0.0139**	**0.0176**	**98.97**

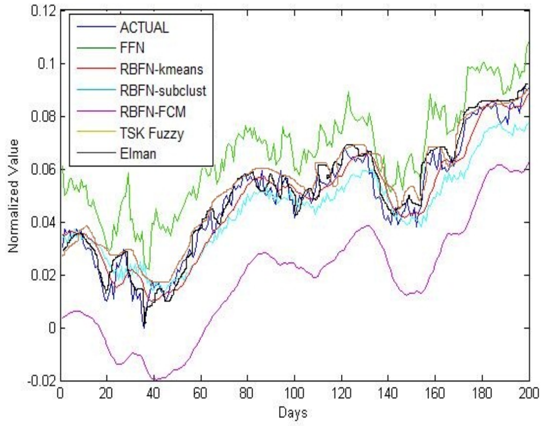

Figure 4.6: Actual Vs. Predicted Values for HDFC Bank

Table 4.2: Comparison of prediction techniques for data of HDFC bank

Prediction technique	RMSE	MMRE	Accuracy
FFNN	0.0162	0.0237	98.63
RBFN with Kmeans Clustering	0.0242	0.0378	97.56
RBFN with Fuzzy C-Means Clustering	0.0301	0.0372	96.46
RBFN with Subtractive Clustering	0.0222	0.0300	97.82
TSK Fuzzy	0.0151	0.0218	98.77
Elman Network	**0.0142**	**0.0208**	**98.85**

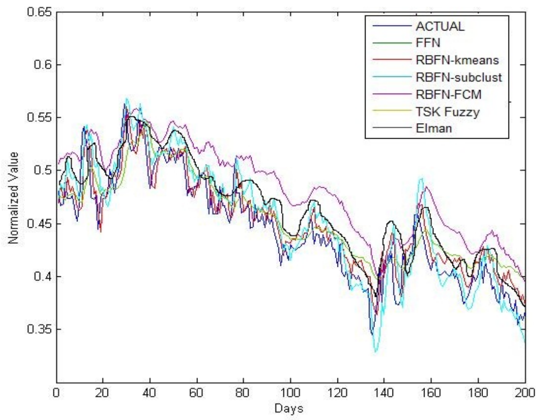

Figure 4.7: Actual Vs. Predicted Values for JSP Ltd.

Table 4.3: Comparison of prediction techniques for data of JSP ltd.

Prediction technique	RMSE	MMRE	Accuracy
FFNN	0.0192	0.0391	98.40
RBFN with Kmeans Clustering	0.0430	0.0913	95.89
RBFN with Fuzzy C-Means Clustering	0.0423	0.0822	96.07
RBFN with Subtractive Clustering	0.0250	0.0408	98.12
TSK Fuzzy	**0.0177**	**0.0251**	**98.84**
Elman Network	0.0283	0.0436	97.63

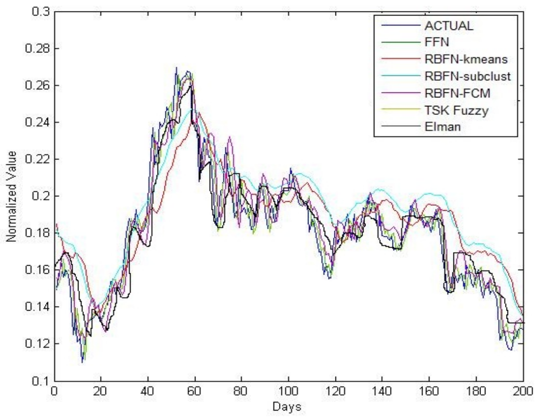

Figure 4.8: Actual Vs. Predicted Values for SAIL

Table 4.4: Comparison of prediction techniques for data of SAIL

Prediction technique	RMSE	MMRE	Accuracy
FFNN	0.0191	0.0480	98.86
RBFN with Kmeans Clustering	0.0204	0.0901	97.07
RBFN with Fuzzy C-Means Clustering	0.0230	0.0878	97.23
RBFN with Subtractive Clustering	0.114	0.0491	98.17
TSK Fuzzy	0.0171	0.0384	98.97
Elman Network	**0.0100**	**0.0233**	**99.32**

Chapter 5

Chapter 5

Conclusion and Future work

Neural networks are well researched and established method that have been used over decades and are very successful in predicting time series behavior from the past datasets. In this report, different neural network techniques and a fuzzy technique have been compared for predicting the stock market price of four organisations. From the analysis, it has been found that from the three RBFN prediction methods each with different clustering method that are used for training the parameters of RBF, the RBFN model with Subtractive clustering, error was less and accuracy was more. RBFN Model with Subtractive clustering when compared with simple FFNN, TSK fuzzy model, and Elman network model, error in Elman network model was least and maximum accuracy for the data of three organisations (SBI, HDFC, SAIL). Only for the dataset of JSP, TSK fuzzy gives least error and maximum accuracy.

In this paper, only the open price has been considered as the inputs and the target data is also the opening price. In the Future work, different factor indicators would be used as input to these models to further improve the accuracy and efficiency of the system and would be compared with other prediction models.

Bibliography

[1] S. J. Taylor, *Modelling financial time series.* No. 2nd, World Scientific Publishing, 2008.

[2] E. E. Peters, *Chaos and order in the capital markets: a new view of cycles, prices, and market volatility.* John Wiley & Sons, 1996.

[3] Y.-F. Wang, "Predicting stock price using fuzzy grey prediction system," *Expert Systems with Applications*, vol. 22, no. 1, pp. 33–38, 2002.

[4] P. Falinouss, "Stock trend prediction using news articles: a text mining approach," 2007.

[5] S. Soni, "Applications of anns in stock market prediction: a survey," *International Journal of Computer Science & Engineering Technology (IJCSET). v2 iMarch (3).*

[6] T. Kimoto, K. Asakawa, M. Yoda, and M. Takeoka, "Stock market prediction system with modular neural networks," in *Neural Networks, 1990., 1990 IJCNN International Joint Conference on*, pp. 1–6, IEEE, 1990.

[7] C. N. Tan and G. E. Wittig, "A study of the parameters of a backpropagation stock price prediction model," in *Artificial Neural Networks and Expert Systems, 1993. Proceedings., First New Zealand International Two-Stream Conference on*, pp. 288–291, IEEE, 1993.

[8] Y. Hiemstra, "A stock market forecasting support system based on fuzzy logic," in *System Sciences, 1994. Proceedings of the Twenty-Seventh Hawaii International Conference on*, vol. 3, pp. 281–287, IEEE, 1994.

[9] H. Tan, D. V. Prokhorov, D. Wunsch, *et al.*, "Conservative thirty calendar day stock prediction using a probabilistic neural network," in *Computational Intelligence for Financial Engineering, 1995., Proceedings of the IEEE/IAFE 1995*, pp. 113–117, IEEE, 1995.

[10] R. J. Kuo, C. Chen, and Y. Hwang, "An intelligent stock trading decision support system through integration of genetic algorithm based fuzzy neural network and artificial neural network," *Fuzzy Sets and Systems*, vol. 118, no. 1, pp. 21–45, 2001.

[11] M. R. Hassan and B. Nath, "Stock market forecasting using hidden markov model: a new approach," in *Intelligent Systems Design and Applications, 2005. ISDA'05. Proceedings. 5th International Conference on*, pp. 192–196, IEEE, 2005.

[12] A. Sheta, "Software effort estimation and stock market prediction using takagi-sugeno fuzzy models," in *Fuzzy Systems, 2006 IEEE International Conference on*, pp. 171–178, IEEE, 2006.

[13] M. R. Hassan, B. Nath, and M. Kirley, "A fusion model of hmm, ann and ga for stock market forecasting," *Expert Systems with Applications*, vol. 33, no. 1, pp. 171–180, 2007.

[14] K. Y. Huang and C.-J. Jane, "A hybrid model for stock market forecasting and portfolio selection based on arx, grey system and rs theories," *Expert systems with applications*, vol. 36, no. 3, pp. 5387–5392, 2009.

[15] C.-H. Cheng, T.-L. Chen, and L.-Y. Wei, "A hybrid model based on rough sets theory and genetic algorithms for stock price forecasting," *Information Sciences*, vol. 180, no. 9, pp. 1610–1629, 2010.

[16] Y.-F. Wang, S. Cheng, and M.-H. Hsu, "Incorporating the markov chain concept into fuzzy stochastic prediction of stock indexes," *Applied Soft Computing*, vol. 10, no. 2, pp. 613–617, 2010.

[17] H.-L. Wong, Y.-H. Tu, and C.-C. Wang, "Application of fuzzy time series models for forecasting the amount of taiwan export," *Expert Systems with Applications*, vol. 37, no. 2, pp. 1465–1470, 2010.

[18] G. S. Atsalakis and K. P. Valavanis, "Forecasting stock market short-term trends using a neuro-fuzzy based methodology," *Expert Systems with Applications*, vol. 36, no. 7, pp. 10696–10707, 2009.

[19] NSE, "Nse - national stock exchange of india ltd.." `http://www.nseindia.com/`.

[20] A. K. Jain, J. Mao, and K. M. Mohiuddin, "Artificial neural networks: A tutorial," *Computer*, vol. 29, no. 3, pp. 31–44, 1996.

[21] D. Svozil, V. Kvasnicka, and J. Pospichal, "Introduction to multi-layer feed-forward neural networks," *Chemometrics and intelligent laboratory systems*, vol. 39, no. 1, pp. 43–62, 1997.

[22] J. J. Moré, "The levenberg-marquardt algorithm: implementation and theory," in *Numerical analysis*, pp. 105–116, Springer, 1978.

[23] A. Ranganathan, "The levenberg-marquardt algorithm," *Tutoral on LM Algorithm*, 2004.

[24] Ö. KISI and E. Uncuoglu, "Comparison of three back-propagation training algorithms for two case studies," *Indian journal of engineering & materials sciences*, vol. 12, no. 5, pp. 434–442, 2005.

[25] H. Yu and B. Wilamowski, "Levenberg-marquardt training," *The Industrial Electronics Handbook*, vol. 5, 2011.

[26] P. D. MILANO, "Clustering - k-means." `http://home.deib.polimi.it/matteucc/Clustering/tutorial_html/kmeans.html`.

[27] P. D. MILANO, "Clustering - fuzzy c-means." `http://home.deib.polimi.it/matteucc/Clustering/tutorial_html/cmeans.html`.

[28] Q. Ren, L. Baron, and M. Balazinski, "Type-2 takagi-sugeno-kang fuzzy logic modeling using subtractive clustering," in *Fuzzy Information Processing Society, 2006. NAFIPS 2006. Annual meeting of the North American*, pp. 120–125, IEEE, 2006.

[29] K. Bataineh, M. Naji, and M. Saqer, "A comparison study between various fuzzy clustering algorithms," *EDITORIAL BOARD*, vol. 5, no. 4, p. 335, 2011.

[30] S. Sivanandam, *Introduction to fuzzy logic using MATLAB.* Springer, 2007.

[31] Z. Liu, X. Wang, J. Xu, L. Cui, and X. Lian, "Prediction technique for water-bloom in lakes based on elman network," in *Automation and Logistics, 2009. ICAL'09. IEEE International Conference on*, pp. 438–444, IEEE, 2009.

[32] A. Vahedi, "The predicting stock price using artificial neural network," *J. Basic. Appl. Sci. Res*, vol. 2, no. 3, pp. 2325–2328, 2012.

[33] H. Wang and Y. Gao, "Elmans recurrent neural network applied to forecasting the quality of water diversion in the water source of lake taihu," in *2010 International Conference on Biology, Environment and Chemistry IPCBEE*, vol. 1, 2011.

Dissemination of Work

Conference:

- Comparison of different Neural Network Models for Stock Market Prediction, *Twenty-Fifth International Conference on Software Engineering and Knowledge Engineering (SEKE 2013)*, June 2013 (Accepted).

- Stock Market Prediction using Neural Network and Fuzzy Logic Models: A Comparative Study, *2nd International Conference on Advances in Computing, Communications and Informatics*, August 2013 (Communicated).